REMARKS

The Examiner has objected to the Specification as failing to provide proper antecedent basis for the claimed subject matter. More specifically, the Examiner has argued that "the phrase 'computer readable medium,' appears to only reasonably convey hardware storage and forms of portable, physical article media to one of ordinary skill in the art." Applicant respectfully disagrees and notes that applicant specifically claims a "computer program product embodied on a tangible computer readable medium" (emphasis added), as claimed. Additionally, applicant notes that the term "tangible computer readable medium" is to be read according to the plain and ordinary meaning thereof, in view of dictionary definitions, and in further view of the definitions provided in the Specification.

Additionally, the Examiner has rejected Claims 1, 3-18, 20-35, 37-47, and 49-54 under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. In the Office Action mailed 05/22/2007, the Examiner has specifically taken issue with the following language as being indefinite: "more strongly." In the Amendment filed 08/22/2007, applicant respectfully asserted that such claim language is to be read according to the plain and ordinary meaning thereof, in view of dictionary definitions, etc. The Examiner, however, has argued that "it is uncertain what the association is stronger than." In response, applicant respectfully asserted that the association is stronger than it would be without the modification of the set of rules.

In the Office Action mailed 11/01/2007, the Examiner has removed the rejection under 35 U.S.C. 112, second paragraph, but has responded to applicant's above arguments. In particular, the Examiner has argued that applicant's above arguments are "not clear from the claim language," and that "it is not clear that the external program calls are more strongly associated with malicious computer program activity as compared to without the modifications." The Examiner has also argued that "[i]t could be more

strongly associated with malicious computer program activity than the primary set of external program calls" such that "the scope of 'more strongly' cannot be ascertained."

Applicant respectfully disagrees. For example, with respect to the independent claims, applicant clearly claims "modifying said set of rules **such that** said at least one secondary set of one or more external program calls are more strongly associated with malicious computer program activity" (see this or similar, but not necessarily identical language in the independent claims-emphasis added), as claimed. Therefore, it is clear that applicant's claimed "said at least one secondary set of one or more external program calls are more strongly associated with malicious computer program activity" (emphasis added), as claimed, is definite.

In the Office Action mailed 07/17/2008, the Examiner has represented the rejection under 35 U.S.C. 112, second paragraph, and has argued that "[t]he term 'more strongly associated' in claims 1, 18, and 35 is a relative term which renders the claim indefinite" and has further argued that "[a]pplicant has failed to provide any actual rationale as to why the claims are definite." Additionally, the Examiner has argued that "it is unclear how modifying 'said set of rules' has any effect on a set of program calls that has already been logged."

Applicant respectfully disagrees. First, applicant again notes that the association of the "at least one secondary set of one or more external program calls" with "malicious computer program activity" is stronger than it would be without the modification of the set of rules, as claimed, which is clearly definite. Additionally, applicant claims "modifying said set of **rules,**" where "a primary set of one or more external program calls matching one or more **rules** indicative of malicious computer program activity from among a set of rules" is "identif[ied] within said stream of external program calls" (see this or similar, but not necessarily identical language in the independent claims-emphasis added), as claimed, which clearly shows the relationship between the "external program calls" and the "modifying," as claimed.

The Examiner has rejected Claims 1, 8-10, 13, 17, 18, 25-27, 30, 34, 35, 42-44, 47, and 51-54 under 35 U.S.C. 102(e) as being anticipated by van der Made (U.S. Patent No. 7,093,239). Applicant respectfully disagrees with such rejection, especially in view of the amendments made hereinabove to each of the independent claims. Specifically, applicant has amended the independent claims to at least substantially include the subject matter of former dependent Claims 53 and 54.

With respect to independent Claims 1, 18 and 35, the Examiner has relied on Col. 6, lines 12-24; and Col. 11, lines 46-60 from the van der Made reference to make a prior art showing of applicant's claimed "secondary set identifying code operable to identify, within said stream, at least one secondary set of one or more external program calls associated with said primary set of one or more external program calls" (see this or similar, but not necessarily identical language in the independent claims).

Applicant respectfully points out that the excerpts from the van der Made reference relied upon by the Examiner merely teach "extracting a behavior pattern and sequence from a modified, new, unknown or suspect program," and that "[t]he behavior pattern is preferably used to analyze the behavior of the unknown program to determine if the behavior of the unknown program is malicious" (Col. 6, lines 13-17 – emphasis added). The excerpts from van der Made also teach that the "ABM engine then analyzes the first executable program and finds that its behavior pattern is altered in a manner indicating that a virus is active" (Col. 11, lines 57-59 – emphasis added).

However, applicant respectfully asserts that only generally disclosing that "[t]he behavior pattern is preferably used to analyze the behavior of the unknown program," as in van der Made, does not specifically meet a "secondary set of identifying code operable to identify, within said stream, at least one secondary set of one or more external program calls associated with said primary set of one or more external program calls" (emphasis added), particularly where the "primary set of one or more external program calls match[es] one or more rules indicative of malicious computer program activity from among a set of rules" (emphasis added), in the context claimed by applicant.

In the Office Action mailed 07/17/2008, the Examiner has argued that "van der Made discloses multiple behavior patterns, where an earlier behavior pattern would be equivalent to the secondary set of programs calls, and a later behavior pattern indicative of a virus infection would be equivalent to the primary set of program calls."

Applicant respectfully disagrees and notes that van der Made merely discloses "detect[ing] malicious code within a computer system by generating and subsequently analyzing a behavior pattern for each computer program introduced to the computer system" and "stor[ing] behavior patterns and sequences with their corresponding analysis results in a database," in addition to disclosing that "[n]ewly infected programs can be detected by analyzing a newly generated behavior pattern for the program with reference to a stored behavior pattern to identify presence of an infection or payload pattern" (Abstract, not specifically cited – emphasis added).

However, merely disclosing the storage of analyzed behavior patterns and the analysis of newly generated behavior patterns with reference to the stored behavior pattern, as in van der Made, fails to disclose "an earlier behavior pattern" and a "later behavior pattern," as argued by the Examiner, and fails to even *suggest* a "secondary set of identifying code operable to identify, within said stream, at least one secondary set of one or more external program calls associated with said primary set of one or more external program calls" (emphasis added), as claimed by applicant.

Furthermore, applicant respectfully points out that detecting active viruses based on whether an executable program's behavior pattern is altered, as in van der Made, clearly fails to teach the use of a "secondary set of identifying code operable to identify, within said stream, at least one secondary set of one or more external program calls associated with said primary set of one or more external program calls" (emphasis added), where the "primary set of one or more external program calls match[es] one or more rules indicative of malicious computer program activity from among a set of rules" (emphasis added), in the context claimed by applicant. Simply nowhere in the excerpts

relied on by the Examiner is there any teaching or suggestion of a "secondary set of one or more external program calls associated with said primary set of one or more external program calls," as claimed.

In the Office Action mailed 11/01/2007, the Examiner has argued that "Made discloses pattern identifying code that can identify program calls associated with malicious activity and are also associated with another set of program calls such as ones that are content destructive since these calls are calls that are made as a result of the first set of calls detected by patterns (6:43-63)."

Applicant respectfully disagrees and asserts that Col. 6, lines 43-63 in van der Made merely discloses that "the analysis procedure specifically targets infection methods such as, but not limited to, the insertion of code to other executables or documents, submitting code to other applications to be transmitted or stored, insertion of code into high memory blocks and the modification of memory control blocks," and that "the analysis method further look[s] for destructive content, such as, but not limited to, functions that overwrite disk areas or the BIOS ROM, or delete files or directories."

Clearly, the excerpts from van der Made merely teach targeting particular infection methods, and separately looking for destructive content, which does not even suggest "identifying code that can identify program calls associated with malicious activity and are also associated with another set of program calls such as ones that are content destructive" (emphasis added), as the Examiner has noted. To this end, the excerpt from van der Made relied on by the Examiner simply does not teach a "secondary set of identifying code operable to identify, within said stream, at least one secondary set of one or more external program calls **associated with said primary set of one or more external program calls**" (emphasis added), where the "primary set of one or more external program calls match[es] one or more rules indicative of malicious computer program activity from among a set of rules" (emphasis added), in the context claimed by applicant.

In the Office Action mailed 07/17/2008, the Examiner has argued that "van der Made specifically discloses that 'generated behavior pattern does not change significantly between version updates, but does change dramatically when a virus infects a program'... [s]ee column 6, lines 30-32." Additionally, the Examiner has argued that "van der Made meets the limitation in question because any one of the behavior patterns logged by the virtual machine that is not considered 'drastically' changed can be considered the claimed 'secondary set of one or more external program calls,' while the 'drastically' changed behavior pattern would be considered the 'primary set.'" Further, the Examiner has argued that "[t]hese patterns are associated to the extent that they are behavior patterns of the same program."

Applicant respectfully disagrees and notes that the excerpt relied on by the Examiner merely discloses that "[t]he generated behavior pattern does not change significantly between version updates, but does change dramatically when a virus infects a program" (Col. 6, lines 30-32). Additionally, applicant notes that van der Made discloses "detect[ing] malicious code within a computer system by generating and subsequently analyzing a behavior pattern for each computer program introduced to the computer system" (Abstract, not specifically cited – emphasis added).

However, merely disclosing the generation and analysis of a behavior pattern for a computer program in a computer system, where a dramatically changed behavior pattern suggests that a program is infected with a virus, as in van der Made, does not disclose the existence of both "drastically" changed and non-"drastically" changed instances of behavior patterns of the same program in a computer system, as suggested by the Examiner, and fails to specifically suggest a "secondary set of identifying code operable to identify, within said stream, at least one secondary set of one or more external program calls **associated with said primary set of one or more external program calls**" (emphasis added), as claimed by applicant.

Still with respect to independent Claims 1, 18 and 35, the Examiner has again relied on Col. 6, lines 12-24; and Col. 11, lines 46-60 from the van der Made reference to

make a prior art showing of applicant's claimed "modifying code operable to modify said set of rules such that said at least one secondary set of one or more external program calls are more strongly associated with malicious computer program activity" (see this or similar, but not necessarily identical language in the independent claims).

Applicant respectfully points out that the excerpts from the van der Made reference relied upon by the Examiner merely teach "extracting a behavior pattern and sequence from a modified, new, unknown or suspect program," and that "[t]he behavior pattern is preferably used to analyze the behavior of the unknown program to determine if the behavior of the unknown program is malicious" (Col. 6, lines 13-17 – emphasis added). Such excerpts from van der Made also teach that the "ABM engine then analyzes the first executable program and finds that its behavior pattern is altered in a manner indicating that a virus is active" (Col. 11, lines 57-59 – emphasis added).

However, applicant respectfully asserts that analyzing "the behavior pattern of the unknown program," and detecting active viruses based on whether an executable program's behavior pattern is altered, as in van der Made, clearly fail to teach "modifying code operable to modify said set of rules such that said at least one secondary set of one or more external program calls are more strongly associated with malicious computer program activity," (emphasis added), as claimed by applicant, particularly where the "rules [are] indicative of malicious computer program activity," in the context claimed. Simply nowhere in the excerpts from the van der Made reference relied on by the Examiner is there any teaching or suggestion to "modify said set of rules," as claimed by applicant.

In the Office Action mailed 11/01/2007, the Examiner has argued that "Made discloses modifying the behavior patterns as new malicious behavior is detected and as more malicious behavior is detected it associated the patterns and the calls that fall within the pattern more closely with the malicious activity (6:25-43)."

Applicant respectfully disagrees and asserts that Col. 6, lines 25-43 in van der Made simply teaches that "a virtual machine is used to generate a behavior pattern and a sequence," and that "[t]he generated behavior pattern does not change significantly between version updates, but does change dramatically when a virus infects a program." However, simply disclosing that a behavior pattern changes when a virus infects a program, as in van der Made, does not even suggest that "as more malicious behavior is detected it associated the patterns and the calls that fall within the pattern more closely with the malicious activity" (emphasis added), as the Examiner has noted. Furthermore, a behavior pattern that changes when a virus infects a program, as in van der Made, does not teach "modifying code operable to modify said set of rules such that said at least one secondary set of one or more external program calls are more strongly associated with malicious computer program activity," (emphasis added), as claimed by applicant, particularly where the "rules [are] indicative of malicious computer program activity," in the context claimed.

In the Office Action mailed 07/17/2008, the Examiner has argued that "van der Made shows (Col. 11, lines 36-60) that the rules used to detect virus behaviors are changed from when an analysis showed [no] virus pattern, to a later analysis that did sho[w] a virus pattern."

Applicant respectfully disagrees and notes that the excerpt relied on by the Examiner merely discloses that "[i]n pre-infection detection, the behavior pattern is analyzed [by the ABM engine] and is found to represent viral behavior for those new or modified programs introduced to the system" and that "[i]n post-infection detection the virus is caught the moment it attempts to infect the first executable on the PC," where "[t]he file hook mechanism detects this attempted change to an executable... [and t]he ABM engine then analyzes the first executable program and finds that its behavior pattern is altered in a manner indicating that a virus is active" (Col. 11, lines 36-60 — emphasis added).

However, merely disclosing the analysis of behavior patterns by an ABM engine in order to find viral behavior, where the analysis is performed on newly introduced programs as well as on programs that pass initial detection but later attempt to change an executable, as in van der Made, does not disclose "that the rules used to detect virus behaviors are changed," as argued by the Examiner, and further fails to disclose "modifying code operable to modify said set of rules such that said at least one secondary set of one or more external program calls are more strongly associated with malicious computer program activity," (emphasis added), as specifically claimed by applicant.

In addition, with respect to independent Claims 1, 18 and 35, the Examiner has relied on Col. 6, lines 12-24 (excerpted below) from the van der Made reference to make a prior art showing of applicant's claimed technique "wherein one of said at least one secondary set of one or more external program calls precedes said primary set of one or more external program calls within said stream of external program calls" (see this or similar, but not necessarily identical language in the independent claims).

> "Preferred implementations of the analytical behavior method (ABM) proceed by extracting a behavior pattern and sequence from a modified, new, unknown or suspect program. The behavior pattern is preferably used to analyze the behavior of the unknown program to determine if the behavior of the unknown program is malicious. Identification of malicious behavior in this manner allows identification of virus carrying files prior to infection of the host computer system. The behavior pattern can also be stored in a database and the virtual machine can subsequently analyze the behavior of the program following modification to determine if its functionality has been modified in a suspect (malicious) manner. This provides post-infection analysis." (Col. 6, lines 12-24 – emphasis added)

Applicant respectfully points out that the excerpt from the van der Made reference relied upon by the Examiner merely teaches "extracting a behavior pattern and sequence from a modified, new, unknown or suspect program," and that "[t]he behavior pattern is preferably used to analyze the behavior of the unknown program to determine if the behavior of the unknown program is malicious" (Col. 6, lines 13-17 – emphasis added).

However, applicant respectfully asserts that only generally disclosing that "[t]he behavior pattern is preferably used to analyze the behavior of the unknown program," as in van der Made, fails to specifically disclose a technique "wherein one of said at least one secondary set of one or more external program calls **precedes** said primary set of one or more external program calls within said stream of external program calls" (emphasis added), as claimed by applicant.

In the Office Action mailed 07/17/2008, the Examiner has argued that "[i]n van der Made, the behavior pattern that did not show a virus pattern would precede the behavior pattern that 'drastically' changed after infection." Applicant respectfully disagrees and again notes that van der Made merely discloses that "[t]he generated behavior pattern does not change significantly between version updates, but does change dramatically when a virus infects a program" (Col. 6, lines 30-32), in addition to disclosing "detect[ing] malicious code within a computer system by generating and subsequently analyzing a behavior pattern for each computer program introduced to the computer system" (Abstract – emphasis added).

However, merely disclosing the generation and analysis of a behavior pattern for a computer program in a computer system, where a dramatically changed behavior pattern suggests that a program is infected with a virus, as in van der Made, does not disclose the existence of both "drastically" changed and non-"drastically" changed instances of behavior patterns of the same program in a computer system, as suggested by the Examiner, and fails to specifically suggest a technique "wherein one of said at least one secondary set of one or more external program calls **precedes** said primary set of one or more external program calls within said stream of external program calls" (emphasis added), as specifically claimed by applicant.

Furthermore, with respect to independent Claims 1, 18 and 35, the Examiner has relied on Col. 11, lines 46-59 (excerpted below) from the van der Made reference to make a prior art showing of applicant's claimed technique "wherein said set of rules is modified to include a new rule corresponding to said secondary set of one or more

external program calls, said new rule thereafter being used in addition to other rules within said set of rules" (see this or similar, but not necessarily identical language in the independent claims).

> "Post-infection detection
>
> Post-infection detection takes place in cases when initial infection is missed by pre-infection detection. A virus could be missed by pre-infection detection when it does not perform any viral function on first execution and does not modify interrupt vectors that point to an infection routine. This is the case with so-called slow infectors and similarly behaving malignant code. In post-infection detection the virus is caught the moment it attempts to infect the first executable on the PC. The file hook mechanism detects this attempted change to an executable (including documents). The AEM engine then analyzes the first executable program and finds that its behavior pattern is altered in a manner indicating that a virus is active." (Col. 11, lines 46-59 - emphasis added)

Applicant respectfully points out that the excerpt from the van der Made reference relied upon by the Examiner merely teaches detecting a virus in an executable program if the program's "behavior pattern is altered in a manner indicating that a virus is active" (Col. 11, lines 58-59 – emphasis added).

However, applicant respectfully asserts that detecting an active virus in a program because the program's behavior pattern is altered, as in van der Made, clearly does not teach that a "set of rules is modified to include a new rule corresponding to said secondary set of one or more external program calls," especially where "said new rule thereafter [is] used in addition to other rules within said set of rules" (emphasis added), as claimed by applicant.

In the Office Action mailed 07/17/2008, the Examiner has again argued that "van der Made shows (Col. 11, lines 36-60) that the rules used to detect virus behaviors are changed from when an analysis showed [no] virus pattern, to a later analysis that did sho[w] a virus pattern."

Applicant respectfully disagrees and again notes that the above excerpt relied on by the Examiner merely discloses that "[i]n pre-infection detection, the behavior pattern is analyzed [by the ABM engine] and is found to represent viral behavior for those new or modified programs introduced to the system" and that "[i]n post-infection detection the virus is caught the moment it attempts to infect the first executable on the PC," where "[t]he file hook mechanism detects this attempted change to an executable... [and t]he ABM engine then analyzes the first executable program and finds that its behavior pattern is altered in a manner indicating that a virus is active" (Col. 11, lines 36-60 – emphasis added).

However, merely disclosing the analysis of behavior patterns by an ABM engine in order to find viral behavior, where the analysis is performed on newly introduced programs as well as on programs that pass initial detection but later attempt to change an executable, as in van der Made, does not disclose "that the rules used to detect virus behaviors are changed," as argued by the Examiner, and further fails to disclose that a "set of rules is modified to include a new rule corresponding to said secondary set of one or more external program calls," especially where "said new rule thereafter [is] used in addition to other rules within said set of rules" (emphasis added), as specifically claimed by applicant.

The Examiner is reminded that a claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described in a single prior art reference. *Verdegaal Bros. v. Union Oil Co. Of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). Moreover, the identical invention must be shown in as complete detail as contained in the claim. *Richardson v. Suzuki Motor Co.*868 F.2d 1226, 1236, 9USPQ2d 1913, 1920 (Fed. Cir. 1989). The elements must be arranged as required by the claim.

This criterion has simply not been met by the above reference excerpt(s), as noted above. Nevertheless, despite such paramount deficiencies and in the spirit of expediting the prosecution of the present application, applicant has amended each of the independent

claims to further distinguish applicant's claim language from the above reference by incorporating the subject matter of former dependent Claims 53 and 54.

With respect to the subject matter of former Claim 53 (now at least substantially incorporated into the independent claims), the Examiner has relied on Col. 10, line 18-Col. 11, line 23; and Col. 12, lines 26-41 from the van der Made reference to make a prior art showing of applicant's claimed "determining whether said modified set of rules decrease malicious network traffic, and promoting said modified set of rules from a temporary set to a permanent set if it is determined that said modified set of rules decrease said malicious network traffic" (see this or similar, but not necessarily identical language in the independent claims).

Applicant respectfully asserts that the excerpts from the van der Made reference relied upon by the Examiner merely teach that "[t]he sequencer contains the order in which the bits were set, identifying the infection sequence shown above" (Col. 10, lines 55-57). Further, the excerpts teach that "[t]he change detection module compares existing files at 6 levels to determine if the file was analyzed previously" (Col. 11, lines 8-9 – emphasis added). Additionally, the excerpts teach that "[i]n tests of a prototype implementation ABM system, the combination of pre-infection (96%) and post-infection detection (4%) resulted in 100% detection of all known viral techniques, using a combination of new, modified and well-known viruses" (Col. 12, lines 26-30 – emphasis added).

However, identifying the infection sequence, comparing files to determine if the file was previously analyzed, and teaching that the combination of pre-infection and post-infection detection resulted in 100% detection of all known viral techniques, as in van der Made, simply fails to suggest "malicious network traffic," much less "determining whether said modified set of rules decrease malicious network traffic, and promoting said modified set of rules from a temporary set to a permanent set if it is determined that said modified set of rules decrease said malicious network traffic" (emphasis added), as claimed by applicant. Clearly, pre-infection and post-infection detection of viral

techniques, in addition to identifying an infection sequence, and determining if a file was previously analyzed, as in van der Made, simply fails to even suggest "**promoting** said modified set of rules from a temporary set to a permanent set if it is determined that said modified set of rules **decrease said malicious network traffic**" (emphasis added), as claimed by applicant.

In the Office Action mailed 07/17/2008, the Examiner has merely argued that "the remaining arguments are fully addressed in light of the above remarks" and has failed to specifically respond to applicant's above arguments with respect to applicant's claimed "**promoting** said modified set of rules from a temporary set to a permanent set if it is determined that said modified set of rules **decrease said malicious network traffic**" (emphasis added), as claimed by applicant. Thus, a notice of allowance or specific prior art showing of each of the foregoing claim elements, in combination with the remaining claimed features, is respectfully requested.

Additionally, with respect to the subject matter of former Claim 54 (now at least substantially incorporated into the independent claims), the Examiner has relied on Col. 12, lines 26-41 from the van der Made reference to make a prior art showing of applicant's claimed "promoting code operable to determine whether said modified set of rules slows malware propagation, and to promote said modified set of rules from a temporary set to a permanent set if it is determined that said modified set of rules slows said malware propagation" (see this or similar, but not necessarily identical language in the independent claims).

Applicant again respectfully asserts that the excerpt from the van der Made reference relied upon by the Examiner merely teaches that "[i]n tests of a prototype implementation ABM system, the combination of pre-infection (96%) and post-infection detection (4%) resulted in 100% detection of all known viral techniques, using a combination of new, modified and well-known viruses" (Col. 12, lines 26-30 – emphasis added).

However, merely teaching that the combination of pre-infection and post-infection detection resulted in 100% detection of all known viral techniques, as in van der Made, simply fails to suggest "promot[ing] said modified set of rules from a temporary set to a permanent set," much less "determin[ing] whether said modified set of rules slows malware propagation, and... promot[ing] said modified set of rules from a temporary set to a permanent set if it is determined that said modified set of rules slows said malware propagation" (emphasis added), as claimed by applicant.

Again, the foregoing anticipation criterion has simply not been met by the above reference excerpt(s), as noted above. Thus, a notice of allowance or specific prior art showing of each of the foregoing claim elements, in combination with the remaining claimed features, is respectfully requested.

Applicant further notes that the prior art is also deficient with respect to the dependent claims. For example, with respect to dependent Claim 7 et al., the Examiner has rejected the same under 35 U.S.C. 103(a) as being unpatentable over van der Made, in view of Obrecht et al (U.S. Patent Publication No. 2004/0064736). More specifically, the Examiner has relied on Paragraph [0039] from the Obrecht reference to make a prior art showing of applicant's claimed technique "wherein score values within a set of rules associated with said secondary set of one or more external program calls are increased to more strongly associate said secondary set of external program calls with malicious computer program activity."

Applicant respectfully notes that the above excerpt from Obrecht relied on by the Examiner merely discloses that "[i]f the result of a malicious code detection routine 54 indicates that the characteristic or behavior of the program being examined was that of a malicious code program, then a weight... is associated with the routine and that weight contributes positively to the malicious code score" (Paragraph [0039]). However, merely associating a weight with a routine if the routine indicates malicious program code behavior, as in Obrecht, fails to disclose a technique "wherein score values within a set of rules associated with said secondary set of one or more external program calls are

increased to more strongly associate said secondary set of external program calls with malicious computer program activity" (emphasis added), as claimed by applicant.

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art and not based on applicant's disclosure. *In re Vaeck*,947 F.2d 488, 20 USPQ2d 1438 (Fed.Cir.1991).

Applicant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the prior art excerpts, as relied upon by the Examiner, fail to teach or suggest all of the claim limitations, as noted above.

Additionally, with respect to Claim 10 et al., the Examiner has relied on Col. 5, lines 16-39 from the van der Made reference to make a prior art showing of applicant's claimed technique "wherein said modifying code dynamically adapts said set of rules in response to detected streams of external program calls performing malicious computer program activity" (see this or similar, but not necessarily identical language in the aforementioned claims).

Applicant respectfully notes that the above reference excerpt relied on by the Examiner merely discloses that "the behavior of a newly loaded or called program is analyzed in a virtual machine that simulates a complete PC... and it is that virtual PC that generates the behavior pattern," where "[t]he virtual PC simulates execution of the new or modified program... and the virtual PC monitors the behavior of the suspect program and makes a record of this behavior that can be analyzed to determine that the target program exhibits virus or malignant behaviors" (Col. 5, lines 16-25 – emphasis added).

Additionally, the excerpts disclose that "[t]he result of the virtual execution by the virtual machine is a behavior pattern representative of the new program," where "the behavior pattern generated by the virtual PC identifies that a program is infected with a virus or is itself a virus" (Col. 5, lines 25-30).

However, merely simulating the execution of a program, monitoring the behavior of the program, and making a record of the program behavior that can be analyzed for virus behavior, as in van der Made, fails to disclose "dynamically adapt[ing] said set of rules in response to detected streams of external program calls performing malicious computer program activity" (emphasis added), as claimed by applicant. Nowhere in the above excerpt is "said set of rules [dynamically adapted] in response to detected streams of external program calls performing malicious computer program activity" (emphasis added), as specifically claimed.

Additionally, with respect to Claim 17 et al., the Examiner has relied on Col. 12, lines 26-41 (excerpted below) from the van der Made reference to make a prior art showing of applicant's claimed technique "wherein said set of rules is subject to a validity check after modification to determine if said set of rules is more effectively detecting malicious computer program activity."

> "In tests of a prototype implementation ABM system, the combination of pre-infection (96%) and post-infection detection (4%) resulted in 100% detection of all known viral techniques, using a combination of new, modified and well-known viruses. Other methods detected only 100% of known viruses and scored as low as 0% for the detection of new, modified and unknown viruses. No exact figure can be quoted for tests involving signature scanner based products. The results for such products are a direct representation of the mix of known, modified and new, unknown viruses; e.g. if 30% of the virus test set is new, modified or unknown then the final score reflected close to 30% missed viruses. No such relationship exists for the implementations of preferred aspects of the present system, where the detection efficiency does not appreciably vary for alterations of the presented virus mix." (Col. 12, lines 26-41 - emphasis added)

Applicant respectfully points out that the excerpt from the van der Made reference relied upon by the Examiner merely discloses "tests of a prototype implementation ABM system, the combination of pre-infection (96%) and post-infection detection (4%) resulted in 100% detection of all known viral techniques, using a combination of new, modified and well-known viruses" (Col. 12, lines 26-41 – emphasis added).

However, applicant respectfully asserts that "tests of a prototype implementation ABM system," as in van der Made, clearly do not teach that a "set of rules is subject to a validity check after modification to determine if said set of rules is more effectively detecting malicious computer activity" (emphasis added), as claimed by applicant. Simply nowhere in the excerpt from the van der Made reference relied on by the Examiner is there any teaching or suggestion of a "validity check after modification [of said set of rules]," as claimed by applicant.

In the Office Action mailed 11/01/2007, the Examiner has argued that "Made discloses a test on a prototype that analyzed the validity of the rules and Made discloses that validity is checked when patterns are detected in order to ensure no false alarms (10:52-11:7)."

> "The resulting behavior pattern is: 24AA952F2A244905
>
> The behavior pattern contains flags that indicate that the user has not had the opportunity to interact with this process through user input (the userInput flag is not set). The sequencer contains the order in which the bits were set, identifying the infection sequence shown above. Therefore this observed behavior is most likely viral.
>
> Many viruses are encrypted, polymorphic or use `tricks` to avoid detection by signature scanners. Wherever such `tricks` are used, the behavior pattern points more obviously towards a virus since such tricks are not normally used in normal applications. In any case, preferred implementations of the present invention require that an infection procedure be present to trigger a virus warning to avoid false positive warnings. Encrypted viruses are no problem, because the execution of the code within the virtual machine, which generates the behavior pattern, effectively decrypts any encrypted or polymorphic virus, as it would in a physical PC environment. Because all parts of the virtual computer are virtualized in preferred embodiments, and at no time is the virtualized program allowed to interact with the physical

> computer, there is no chance that viral code could escape from the virtual machine and infect the physical computer." (Col. 10, line 52 - Col. 11, line 7 - emphasis added).

Applicant respectfully disagrees and asserts that the excerpt from the van der Made reference relied upon by the Examiner merely teaches that "[t]he behavior pattern contains flags that indicate that the user has not had the opportunity to interact with this process through user input" and that "preferred implementations of the present invention require that an infection procedure be present to trigger a virus warning to avoid false positive warnings" (emphasis added).

However, teaching that the behavior pattern contains flags indicating that the user has not had the opportunity to interact with this process, in addition to teaching that an infection procedure is required to be present to trigger a virus warning, as in van der Made, simply fails to suggest that a "set of rules is subject to a validity check after modification to determine if said set of rules is more effectively detecting malicious computer activity" (emphasis added), as claimed by applicant. Clearly, a flag in the behavior pattern indicating that the user has not interacted with the process, as in van der Made, simply fails to even suggest that a "set of rules is subject to a validity check after modification" (emphasis added), as claimed by applicant.

In the Office Action mailed 07/17/2008, the Examiner has argued that "van der Made effectively shows that changed behavior analysis detected 4% of the virus that the initial analysis did not detect, and therefore meets the claim limitation." Applicant respectfully disagrees and again notes that van der Made merely discloses "tests of a prototype implementation ABM system," where the ABM engine analyzes behavior patterns in order to find viral behavior, and where the analysis is performed on newly introduced programs as well as on programs that pass initial detection but later attempt to change an executable.

However, merely testing an implementation of a system that analyzes behavior patterns of both newly introduced programs as well as programs that pass initial detection

but later attempt to change an executable, as in van der Made, clearly does not teach that a "set of rules is subject to a validity check after modification to determine if said set of rules is more effectively detecting malicious computer activity" (emphasis added), as claimed by applicant. Simply nowhere in the excerpt from the van der Made reference relied on by the Examiner is there any teaching or suggestion of a "validity check after modification [of said set of rules]," as claimed by applicant.

Additionally, with respect to Claim 52, the Examiner has relied on Col. 10, line 18-Col. 11, line 23; and Col. 12, lines 26-41 from the van der Made reference to make a prior art showing of applicant's claimed "applying high level rules to said modified set of rules, and promoting said modified set of rules from said temporary set to said permanent set based on the application of the high level rules to said modified set of rules" (as amended).

Applicant respectfully asserts that the excerpts from the van der Made reference relied upon by the Examiner merely teach that "[t]he sequencer contains the order in which the bits were set, identifying the infection sequence shown above" (Col. 10, lines 55-57 – emphasis added). Further, the excerpts teach that "[t]he change detection module compares existing files at 6 levels to determine if the file was analyzed previously" (Col. 11, lines 8-9 – emphasis added). Additionally, the excerpts teach that "[i]n tests of a prototype implementation ABM system, the combination of pre-infection (96%) and post-infection detection (4%) resulted in 100% detection of all known viral techniques, using a combination of new, modified and well-known viruses" (Col. 12, lines 26-30 – emphasis added).

However, identifying the infection sequence, comparing files to determine if the file was previously analyzed, and teaching that the combination of pre-infection and post-infection detection resulted in 100% detection of all known viral techniques, as in van der Made, simply fails to suggest "applying high level rules to said modified set of rules, and promoting said modified set of rules from said temporary set to said permanent set based on the application of the high level rules to said modified set of rules" (emphasis added),

as claimed by applicant. Clearly, pre-infection and post-infection detection of viral techniques, in addition to identifying an infection sequence, and determining if a file was previously analyzed, as in van der Made, simply fails to even suggest "**promoting** said **modified** set of rules from said **temporary set** to said **permanent set** based on the application of the high level rules to said modified set of rules" (emphasis added), as claimed by applicant.

In the Office Action mailed 07/17/2008, the Examiner has merely argued that "the remaining arguments are fully addressed in light of the above remarks" and has failed to specifically respond to applicant's above arguments with respect to applicant's claimed "**promoting** said **modified** set of rules from said **temporary set** to said **permanent set** based on the application of the high level rules to said modified set of rules" (emphasis added), as claimed by applicant. Thus, a notice of allowance or specific prior art showing of each of the foregoing claim elements, in combination with the remaining claimed features, is respectfully requested.

Again, with respect to the rejection under 35 U.S.C. 102(e), since the above anticipation criterion has simply not been met by the above reference excerpt(s), as noted above, a notice of allowance or specific prior art showing of each of the foregoing claim elements, in combination with the remaining claimed features, is respectfully requested.

Additionally, with respect to the rejection under 35 U.S.C. 103(a), since at least the third element of the *prima facie* case of obviousness has not been met, a notice of allowance or specific prior art showing of each of the foregoing claim elements, in combination with the remaining claimed features, is respectfully requested.

Still yet, applicant brings to the Examiner's attention the subject matter of new Claim 55 below, which is added for full consideration:

"wherein one or more higher-level rules are applied to said modified set of rules to determine if said modified set of rules is more effectively detecting malicious computer program activity after modification" (see Claim 55).

Again, a notice of allowance or a proper prior art showing of all of applicant's claim limitations, in combination with the remaining claim elements, is respectfully requested.

Thus, all of the independent claims are deemed allowable. Moreover, the remaining dependent claims are further deemed allowable, in view of their dependence on such independent claims.

In the event a telephone conversation would expedite the prosecution of this application, the Examiner may reach the undersigned at (408) 505-5100. The Commissioner is authorized to charge any additional fees or credit any overpayment to Deposit Account No. 50-1351 (Order No. NAI1P489).

Respectfully submitted,
Zilka-Kotab, PC

/KEVINZILKA/

Kevin J. Zilka
Registration No. 41,429

P.O. Box 721120
San Jose, CA 95172-1120
408-505-5100